IN THE CLAIMS

1. (Currently Amended) A method for recovering from data errors within a processor, comprising the steps of:

for each cycle of the processor, storing a copy of data from at least one, but not all, registers of a register file within a buffer ~~prior to architecting new data in the register~~ if new data is architected to the registers and if the cycle is not a checkpoint cycle;

~~periodically~~ checking for data errors within the processor if the cycle is a checkpoint cycle; and

restoring the data from the buffer to the register file in the event of data errors.

2. (Canceled)

3. (Currently Amended) A method of claim 1, further comprising loading the new data to the registers after the step of storing.

4. (Currently Amended) A method of claim 1, further comprising loading the new data to the registers concurrently with the step of storing.

5. (Original) A method of claim 1, the step of storing the data within the buffer comprising storing the data within a second register file.

6. (Original) A method of claim 1, further comprising the step of flushing the buffer after checking for, and detecting no, data errors.

7. (Original) A method of claim 1, further comprising the step of freezing execution of instructions within pipelines of the processor after detecting data errors.

8. (Previously Presented) A method of claim 1, further comprising the step of resetting a program counter of the processor after detecting errors.

9. (Previously Presented) A method of claim 8, further comprising a step of re-executing a program through the processor at a time associated with the reset program counter.

10. (Currently Amended) A method of claim 1, the step of ~~periodically~~ checking for data errors comprising periodically checking for the data errors at sequential time periods defined by a number of processor clock cycles.

11. (Original) A method of claim 1, further comprising the steps of utilizing an error correction code in connection with data storage to the buffer.

12. (Currently Amended) A processor with register file data recovery, comprising:

an execution unit having a plurality of pipelines for processing program instructions relative to a program counter;

a register file, wherein one or more stages of the pipelines loads new data to ~~a~~ one or more registers of the register file; and

a buffer for storing a copy of data within ~~the~~ at least one, but not all, registers prior to loading the new data, and for restoring data to the register file in the event ~~of~~ data errors are detected at a checkpoint within the processor;

wherein the buffer is flushed at the checkpoint if no data errors are detected and wherein the checkpoint occurs each plurality of processor cycles.

13. (Original) A processor of claim 12, the buffer comprising a second register file.

14. (Original) A processor of claim 12, the register file comprising an extra read port for reading the data from the register.

15. (Original) A processor of claim 12, the register file comprising a write port for writing the data from the buffer to the register.

16. (Original) A processor of claim 12, further comprising one or more error detectors for detecting the data errors.

17. (Original) A processor of claim 16, the error detectors comprising redundant logic devices.

18. (Original) A processor of claim 12, further comprising error correction code for data recovery of data stored within the buffer.

19. (Currently Amended) A processor of claim 12, the buffer reading data within the registers prior to an execution stage for an instruction identifying a write to the registers.

20. (Currently Amended) A processor of claim 12, ~~wherein~~ the program counter being reset in connection with the buffer restoring data to the register file.

21. (New) A method of claim 1, the step of checking comprising checking for data errors within the processor each plurality of cycles.